U.S. DEPARTMENT OF COMMERCE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
NATIONAL WEATHER SERVICE
SYSTEMS DEVELOPMENT OFFICE
TECHNIQUES DEVELOPMENT LABORATORY

TDL Office Note 75-2

COMPUTER PROGRAMS FOR THE MOS DEVELOPMENT SYSTEM
IBM 360/195 VERSION

Edited by

Harry R. Glahn, George W. Hollenbaugh
and
Frank T. Globokar

February 1975

CONTENTS

Computer Programs for the MOS Development System
IBM 360/195 Version

Edited by

Harry R. Glahn, George W. Hollenbaugh
and
Frank T. Globokar

INTRODUCTION

Model Output Statistics (MOS) is a technique in which a predictand is related statistically to predictors which are themselves forecasts (or output) from numerical models. A comprehensive set of IBM 360/195 computer programs is available for processing the extensive data collections available within TDL. These data collections are a part of the MOS development system described by Glahn in TDL Office Note 74-14. Writeups of those programs which are used within TDL for developing MOS products are provided in this Note.

Our library of MOS-related programs (MOSLIB) reside on the IBM 360/195 load module NWS.SDO.TDL.LOAD. Will Shaffer and Farnese Hicks handle the housekeeping functions on this data set. The librarian for those programs related to the MOS development system is George Hollenbaugh. If someone wishes to add a program to MOSLIB, he should prepare a writeup in the format of the writeups contained in this Note and submit it to the librarian. After the writeup is approved, the librarian will put the program on the load module with a member name the same as the program name prefixed with the letters "OP". (For instance, subroutine YPKR has a member name OPYPKR.) No member that has a name with the prefix "OP" may be changed without permission from the MOSLIB librarian. The librarian will distribute the program writeup, which can be inserted in this loose-leaf Note. The librarian will also maintain a backup deck and a compilation listing of all MOSLIB programs for easy access and reference.

YPKR

PACKS MOS SYSTEM PREDICTAND TAPE RECORDS

Harry R. Glahn
July 6, 1974
December 31, 1974 (Rev.)

PURPOSE: The MOS System predictand data matrix DATA(,) is packed by column into NPK( ) starting at location 2 and the date NDATE1 inserted into NPK(1). NPK( ) is then in the format prescribed in "Format of MOS Predictand Tapes".

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
DIMENSION NPK(1410),DATA(260,17)
.
CALL YPKR(NPK,NWDS,NROWS,NCOLS,DATA,MD1,MD2,NDATE1,M)
.
```

NPK( ) - Packed data and date returned to calling program (see "Format of MOS Predictand Tapes"). Dimension is not necessarily 1410, but is recommended for consistency among MOS programs. This allows for a maximum of 260 stations and 18 columns.

NWDS - Dimension of NPK( ).

NROWS - Number of rows (first dimension) of data in DATA(,).

NCOLS - Number of columns (second dimension) of data in DATA(,). Must not exceed 18.

DATA(,) - Predictand data matrix supplied to YPKR for packing. Dimensions are not necessarily 260 and 18 but are recommended for consistency among MOS programs.

MD1 - First dimension of DATA(,).

MD2 - Second dimension of DATA(,).

NDATE1 - Date in (integer) form YR*1000000 + MO*10000 + DA*100 + HR

M - Words in NPK( ) filled after packing--returned to calling program

YUNPKR

UNPACKS MOS SYSTEM PREDICTAND TAPE RECORDS

Harry R. Glahn
July 6, 1974
December 31, 1974 (Rev.)

PURPOSE: One column of the MOS System predictand data matrix, which is furnished to YUNPKR in packed form in NPK( ) (see "Format of MOS Predictand Tapes"), is unpacked and returned in VRBL( ).

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
DIMENSION NPK(1410),VRBL(260)
.
CALL YUNPKR(NPK,NWDS,NROWS,NCOLS,VRBL,NPOS)

.
```

NPK( ) - Packed data and date furnished to subroutine. (See "Format of MOS Predictand Tapes".) Dimension is not necessarily 1410 but is recommended for consistency among MOS programs. This allows for a maximum of 260 stations and 18 columns.

NWDS - Dimension of NPK( ).

NROWS - Number of items to unpack = number of rows (first dimension) of original data matrix.

NCOLS - Number of types of data in NPK( ) = number of columns (second dimension) of original data matrix.

VRBL( )- Unpacked data are returned in VRBL( ). Dimension is not necessarily 260, but is recommended for consistency among MOS programs.

NPOS - The portion of NPK( ) which corresponds to column NPOS in the original data matrix is unpacked and stored in VRBL( ). Maximum value = NCOLS.

EXAMPLE: CALL YUNPKR(NPK, 1410, 255, 18, VRBL, 4)

The portion of NPK(1410) which corresponds to the 4th column (second dimension) in the original data matrix DATA(,) is unpacked and stored in VRBL(i), i=1,255. Original data

M430

CONVERTS CDC 6600 MOS PREDICTAND TAPES TO IBM 360/195

Harry R. Glahn
July 15, 1974

PURPOSE: MOS predictand data exist in packed form on CDC 6600 tapes (see "Format of MOS Predictand Tapes"). A tape is read by M430 and written for use on the IBM 360/195. The same record structure is maintained (see "Predictand Data Matrix Description").

CARD INPUT: None.

TAPE INPUT: Data Set Reference Number 1

A single 7-track, 800 bpi, binary mode tape, written with the FORTRAN WRITE statement and the L-Tape driver on the CDC 6600 is read with the subroutine READ7T. This tape contains one or more files of predictand data seperated by CDC 6600 "software" EOF's.

TAPE OUTPUT: Data Set Reference Number 2

A 9-track, 1600 bpi, binary mode tape is written with the asychronous FORTRAN WRITE statement. Each file of data on the input tape becomes a "pseudo-file" of data on the output tape, ending with a dummy record (see "Format of MOS Predictand Tapes").

OUTPUT: Printout consists mainly of the first 7 header records and the first data record in each "pseudo-file" of data.

RESTRICTIONS: M430 was written to process the CDC 6600 data tapes in existence at the time of writing. Any other use of the program is questionable.

NONSYSTEM ROUTINES CALLED: W3AI08 and YUNPKR.

STORAGE REQUIREMENTS: 400K is sufficient.

LANGUAGE: FORTRAN IV (H Extended).

LOCATION: Exists on load module DSN=NWS.SDO.TDL.LOAD with member name OPM430.

M640

PLOTS A VARIABLE FROM MOS PREDICTAND TAPE

Thomas D. Bethem
November 20, 1974

PURPOSE: Plots a variable read from a MOS predictand tape for specific dates for all conterminous U.S. stations.

CARD INPUT: Card order must be followed as given. All integers are right adjusted.

Card Type 1 - Format (3I4)

NPOS - Position (column) in predictand data matrix of data to plot (see Predictand Data Matrix Description).

MYTAU - Hour added to date in NDATE( ) to indicate record on predictand tape--must be 24 or less.

NBK - Indicates type of map background desired: 0=none, 1=U.S. outline only, 2=U.S. outline and state boundaries.

Card Type 2 - Format (10A8)

TAPS(2,) - List of predictand tape reel numbers. A blank word is the terminator unless exactly 10 tapes are used, in which case the card is full and no terminator is required. Maximum tapes per run is 10. This list is for printout only; actual tape selection is made from JCL DD cards.

Card Type 3 - Format (20A4)

TITLE( ) - 80-character title to be printed on each map.

Card Type 4 - Format (9I8)

NDATE( ) - All dates are read and placed into NDATE( ). Blanks and zeros on the card are permissible. The terminator is 999999. Dates must be in increasing order and are of form: 2 digits of the year X 10000 + the number of the month X 100 + the day of the month. Maximum of 63 dates can be used.

Card Type 5 - Format (12I6)

JSTA( ) - 5-digit WBAN numbers of stations for which data are to be plotted. Blank and zero words are permissible. Terminator is 999999. Maximum of 234 stations can be used.

NONSYSTEM ROUTINES CALLED: RDXY, RDY1, RDY2, RDY3, RDLIST, YUNPKR, CHNGDT, and MAPS.

STORAGE REQUIREMENTS: 120K is sufficient.

LANGUAGE: FORTRAN IV (H Extended).

LOCATION: Exists on load module DSN=NWS.SDO.TDL.LOAD with member name OPM640. All subroutines are on the same module with member names the same as the subroutine names with an "OP" prefix (see example JCL).

EXAMPLE JCL:

```
//MM640 JOB (ME2000BC2940420,GRAMX),'BETHEM TDL',
//   REGION=120K,TIME=1,MSGLEVEL=(1,1)
// EXEC MEORXLG
//LKED.MYLIB DD DSN=NWS.SDO.TDL.LOAD,DISP=SHR
//LKED.SYSIN DD *
       INCLUDE MYLIB(OPMAPS,OPRDXY,OPRDY1,OPRDY2,OPRDY3,OPYUNPKR)
       INCLUDE MYLIB(OPCHNGDT,OPRDLIST,OPM640)
       ENTRY MAIN
//GO.FT02F001 DD DCB=(RECFM=VS,LRECL=5204,DEN=3),DISP=(OLD,KEEP,KEEP),
//            DSN=YDATA,LABEL=(,SL,,IN),UNIT=TAPE9,VOL=SER=E10347
//GO.SYSIN DD *
                INPUT CARDS
/*
//
```

RDX

READS MOS SYSTEM INTERPOLATED PREDICTOR TAPES

Harry R. Glahn
August 15, 1974
December 31, 1974 (Rev.)

PURPOSE: Packed and blocked interpolated predictor tapes (see "Format of MOS Interpolated Predictor Tapes") are read and a specified field is returned unpacked to the calling program, or, optionally, the identification of the next field encountered for the desired date is returned. Tapes are changed by the 360/195 system when an EOF is encountered, and a message to that effect is printed by the subroutine RDXY. User is not allowed to accept data on which a reading error has occurred but does have the option of skipping records. The first time RDX is called, only header information is read and initialization done.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
DIMENSION XDATA(260),MPRED1(2),MPRED2(2),NWD(4),JSTA(260),LOCX(260),
     NAMEX(260,5),LWBANX(260)
COMMON/BLOCK4/NTAPS(2),TAPS(2,16)
REAL*8 TAPS
.
CALL RDX(ITAPEX,X,X,X,X,X,X,X,NSTA,JSTA,LOCX,NAMEX,LWBANX,NEOFX,ND3)
.
CALL RDX(ITAPEX,MDATE,MPRED1,XDATA,MSRCH,NSTATS,MPRED2,NWD,NSTA,JSTA,
     LOCX,NAMEX,LWBANX,NEOFX,ND3)
.
```

ITAPEX - Logical unit number of tape from which to read data. (For consistency among MOS programs, use 1 if possible.)

MDATE - Date for which data are desired in (integer) form YR*1000000 + MO*10000 + DA*100 + HR. The date is the initial or "run" time and does not include the "tau" or projection. In successive calls to RDX, MDATE must be non-decreasing, unless tape is rewound and NEOFX set = 0.

MPRED1(J),J=1,2 - Predictor being sought in hex form yyyxxx (J=1) and mmffttt (J=2) if MSRCH not zero (see "Format of MOS Interpolated Tapes"). Not used if MSRCH=0. In successive calls to RDX, the predictors must be asked for in the order in which they exist on the tape.

NOTE: The first time RDX is called, only header records are read, variables LOCX( ), NAMEX(,), LWBANX( ) initialized, and NEOFX set =1. Dummy arguments can be used as indicated by "X" in the examples.

LABELED COMMON VARIABLES:

NTAPS(1) - Normally contains the number of predictor tapes available.

LTAPS(1,J),J=1,NTAPS(1) - Tape reel numbers of tapes available in EBCDIC, maximum of 16. Variable is REAL*8.

EXAMPLE:

```
DIMENSION XDATA(260),MPRED1(2),MPRED2(2),NWD(4),JSTA(260),LOCX(260),
     NAMEX(260,5),LWBANX(260)
DATA MPRED1/Z240001,Z000024/
NTAPS(1) = 2
LTAPS(1,1) = ' E10356 '
LTAPS(1,2) = ' E10357 '
NSTA = 2
JSTA(1) = 3103
JSTA(2) = 3812
NEOFX = 0
ND3 = 260
CALL RDX(1,X,X,X,X,X,X,X,X,NSTA,JSTA,LOCX,NAMEX,LWBANX,NEOFX,ND3)
CALL RDX(1,72100112,MPRED1,XDATA,1,NSTATS,MPRED2,NWD,NSTA,JSTA,LOCX,
     NAMEX,LWBANX,NEOFX,ND3)
```

Unmodified 36-hr 1000-mb forecast heights from the 1200 GMT PE run for October 1, 1972 are to be returned in XDATA( ) with tape-reading status in NSTATS. Two tapes, E10356 and E10357, are available for use. Data from two stations, 3103 (Flagstaff, Ariz.) and 3812 (Asheville, N.C.), are to be used in the calling program. (Tape numbers cannot actually be specified as shown above; they can be read from cards or provided in a DATA statement in a BLOCK DATA subprogram.) If MSRCH = 0, RDX must be called again through the entry point XUNPKR to secure unpacked data. An alternative to the above single call to RDX is:

```
CALL RDX(1,72100112,X,XDATA,0,NSTATS,MPRED2,NWD,NSTA,JSTA,LOCX,NAMEX,
       LWBANX,NEOFX,ND3)
If (MPRED2(1).EQ.MPRED1(1).AND.MPRED2(2).EQ.MPRED1(2)) CALL XUNPKR
       (X,X,X,XDATA,X,X,X,NWD,X,X,X,X,X,X,ND3)
```

Of course, NSTATS must always be checked before data are used.

OUTPUT: Printout is limited to diagnostic information.

RESTRICTIONS: LABELED COMMON block BLOCK4 is used by RDX.

NONSYSTEM ROUTINES CALLED: RDXY and W3AI01.

STORAGE REQUIREMENTS: 2046 hexadecimal bytes.

LANGUAGE: Fortran IV (H Extended).

LOCATION: RDX and RDXY exist on load module DSN=NWS.SDO.TDL.LOAD with member names OPRDX and OPRDXY, respectively. W3AI01 is on load module DSN=NWS.NMC.TEST.LOAD with member name W3AI01.

COMMENTS: The loss of accuracy through packing and unpacking is not a problem except when it is desired to recover a particular value exactly. RDX assumes that all zeros in precipitation amount fields (first 3 digits of hex identifier = 132) were set = a negative number (-.00508 for PE and LFM, and -.0001 for Trajectory model) before packing by W3AI00 and arranges to return exact zeros to calling program. Unpacking is done with W3AI01.

RDY1

READS MOS SYSTEM PREDICTAND TAPES

Harry R. Glahn
August 15, 1974
December 31, 1974 (Rev.)

PURPOSE: Reads packed predictand data records (see "Format of MOS Predictand Tapes") and returns a single unpacked variable in VRBL( ) from data matrix column NPOS. Tapes are changed by the 360/195 system when an EOF is encountered, and a message to that effect is printed by subroutine RDXY. User is not allowed to accept data on which a reading error has occurred but does have the option of skipping those records. The first time RDY1 is called only header information is read and initialization done.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
DIMENSION VRBL(260),JSTA(260),LOCY(260),NAMEY(260,5),LWBANY(260)
COMMON/BLOCK4/NTAPS(2),TAPS(2,16)
.
REAL*8 TAPS
.
CALL RDY1(ITAPEY,X,X,X,X,X,NSTA,JSTA,LOCY,NAMEY,LWBANY,NEOFY,ND3)
.
CALL RDY1(ITAPEY,NDATE1,MYTAU,NPOS,VRBL,NSTATS,NSTA,JSTA,LOCY,
    NAMEY,LWBANY,NEOFY,ND3)
.
```

ITAPEY - Logical unit number of tape on which predictand data reside. (For consistency among MOS programs, use 2 if possible.)

NDATE1 - Date in (integer) form YR*1000000 + MO*10000 + DA*100 + HR (see MYTAU).

MYTAU - Number of hours to add to NDATE1 to give data and time for which data are desired. When the last two digits of the date-time exceed 24, changing of day, month, and year are provided for. Usually, NDATE1 will be the date-time of the model run, called the basic date, and MYTAU provides the "lag" for which predictand data are required. In successive calls to RDY1, the dates specified must be non-decreasing, unless tape is rewound and NEOFY set = 0.

LABELED COMMON VARIABLES:

NTAPS(2) - Normally contains the number of predictand tapes available.

TAPS(2,J),J=1,NTAPS(2) - Tape reel numbers of predictand tapes available in EBCDIC, maximum of 16. Variable is REAL*8.

EXAMPLE:

```
DIMENSION VRBL(260),JSTA(260),LOCY(260),NAMEY(260,5),LWBANY(260)
NTAPS(2) = 2
TAPS(2,1) = ' E10358 '
TAPS(2,2) = ' E10359 '
NSTA = 2
JSTA(1) = 3103
JSTA(2) = 3812
NEOFY = 0
ND3 = 260
CALL RDY1(2,X,X,X,X,X,NSTA,JSTA,LOCY,NAMEY,LWBANY,NEOFY,ND3)
CALL RDY1(2,72100112,36,4,VRBL,NSTATS,NSTA,JSTA,LOCY,NAMEY,LWBANY,NEOFY,ND3)
```

The fourth column (coded total sky cover) of the data matrix for 0000 GMT October 3, 1972 (1200 GMT October 1 plus 36 hours) is returned unpacked in VRBL( ) from logical unit 2. Two tapes, E10358 and E10359, are available for use. Data from two stations, 3103 (Flagstaff, Ariz.) and 3812 (Asheville, N.C.) are to be used in the calling program. (Tape numbers cannot actually be specified as shown above; they can be read from cards or provided in a DATA statement in a BLOCK DATA sub-program.) Of course, NSTATS must always be checked before data are used.

OUTPUT: Printout is limited to diagnostic information.

RESTRICTIONS:

LABELED COMMON block BLOCK4 is used by RDY1. Program saves last date read in JYDATE. If the date desired, NDATE1 + MYTAU, is previous to JYDATE, NSTATS is set = 2 and control is returned to the calling program. To circumvent this, JYDATE must be reinitialized. This can be done only by setting NEOFY = 0. This assumes a restart and tape must be rewound.

NONSYSTEM ROUTINES CALLED: RDY2, RDY3, RDXY, CHNGDT and YUNPKR.

STORAGE REQUIREMENTS: 2032 hexadecimal bytes.

LANGUAGE: Fortran IV (H Extended).

LOCATION: RDY1, RDY2, RDY3, RDXY, CHNGDT, and YUNPKR exist on load module DSN=NWS.SDO.TDL.LOAD with member names OPRDY1, OPRDY2, OPRDY3, OPRDXY, OPCHNGDT, AND OPYUNPKR, respectively.

RDY2

READS VARIABLE LENGTH UNFORMATTED RECORD

Harry R. Glahn
August 15, 1974

PURPOSE: Reads a variable length unformatted record with the FORTRAN READ statement READ(ITAPE) DATA. The dimension of DATA( ) is transmitted to RDY2 as a parameter in the calling sequence. RDY2 probably performs this data transfer more efficiently than the statement READ(ITAPE)(DATA(I),I=1,NSIZE).

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
DIMENSION DATA( )
.
CALL RDY2(ITAPE,DATA,NSIZE,INDEX)
.
```

ITAPE - Data set reference number of tape from which to read data.

DATA( )- Data are returned to calling program in DATA( ).

NSIZE - Number of words to read from the next record on ITAPE.

INDEX - Tape reading status returned to calling program:

0 = End of file.
1 = Reading error.
2 = Good read.

EXAMPLE:

```
DIMENSION DATA (260)
CALL RDY2(2,DATA,100,INDEX)
```

The first 100 words from the next record on reference number 2 (number 2 can be a tape or system disk file) are read and returned in DATA.

STORAGE REQUIREMENTS: 192 hexadecimal bytes.

LANGUAGE: Fortran IV (H Extended).

COMMENTS: RDY2 was written to be used by RDY1, but is completely general and can be used by other programs.

MOS OP NO. 8

CHNGDT

ADDS HOUR TO DATE-TIME AND PUTS RESULT IN MOS FORMAT

Harry R. Glahn
August 15, 1974

PURPOSE: Adds a number of hours (KXTAU) to a date-time in basic MOS format and modifys hour, day, month, and year as necessary so that the resulting date-time contains an hour in the range 1 to 24 inclusive.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
CALL CHNGDT(NDATE,KXTAU,NDATE1)
.
```

NDATE - Date-time in MOS format YR*1000000 + MO*10000 + DA*100 + HR.

KXTAU - A number of hours to add to HR in NDATE. Can be negative.

NDATE1 - Date-time in MOS format returned to calling program after KXTAU has been added to HR and HR,DA,MO, and YR adjusted as necessary so that $0<HR\leq 24$.

EXAMPLE: CALL CHNGDT(72123012,48,NDATE1)

NDATE1 is returned as 73010112.

RESTRICTIONS: None.

NONSYSTEM ROUTINES CALLED: None.

STORAGE REQUIREMENTS: 378 hexadecimal bytes.

LANGUAGE: Fortran IV (H Extended).

LOCATION: On load module DSN=NWS.SDO.TDL.LOAD with member name OPCHNGDT.

M600

PERFORMS SCREENING REGRESSION ANALYSIS

Harry R. Glahn
December 31, 1974

PURPOSE: This screening regression subroutine is designed specifically for use with the TDL MOS data collection. Screening is performed in the forward stepwise manner according to the least squares criterion. Predictors can be binary (cumulative dummies) and/or continuous. If desired, the predictand(s) can be divided into categories for probability estimation. M600 utilizes variable dimensions so that storage requirements can be altered as needed. A driver DRM600 is available in source form.

CALL AND EXPLANATION OF DRIVER DRM600

```
C     PROGRAM DRM600
C        FEBUARY 1974  GLAHN      IBM 360/195
C          PURPOSE
C            DRIVER FOR M600 REGRESSION SUBROUTINE
C            SEE M600 COMMENT CARDS OR M600 WRITEUP FOR EXPLANATIONS
C          LET   ND1=THE MAXIMUM NUMBER OF DATES THAT CAN BE USED.
C                ND2=THE NUMBER OF POSSIBLE PREDICTORS PLUS THE NUMBER OF
C                    PREDICTAND CATEGORIES.  MUST BE 4 OR GREATER.
C                ND3=THE MAXIMUM NUMBER OF ROWS IN THE PREDICTAND MATRIX
C                    OR THE NUMBER OF INTERPOLATED VALUES (STATIONS) ON
C                    THE PREDICTOR TAPE, WHICHEVER IS LARGER. NORMALLY,
C                    THIS SHOULD BE 260 FOR ASHVILLE DATA USERS (FOR
C                    CONSISTENCY) AND 869 FOR THE MDR-RELATED RUNS.
C                ND4=THE NUMBER OF STATIONS TO BE USED, EITHER FOR
C                    SINGLE STATION OR GENERALIZED EQUATIONS.
C                ND5=((ND2*ND2+ND2)/2)+ND2+1
C                ND6=AN INTEGER EQUAL TO OR GREATER THAN 1 AND EQUAL TO
C                    OR LESS THAN NEQ, THE NUMBER OF EQUATION SETS TO BE
C                    DEVELOPED.
C          THEN  IT IS SUFFICIENT FOR DIMENSIONS OF VARIABLES TO BE
C                 AS FOLLOWS (EXCEPT DIMENSIONS OF P MUST BE
C                 (4,9) OR GREATER)
C                NDATE(ND1)
C                KXTAU(ND2),NCAT(ND2),CAT(ND2),AVG(ND2),SIG(ND2),S(ND2),
C                     AATMP(ND2)
C                NPRED(2,ND2)
C                NWORD(4,ND2)
C                NGP(ND4)
C                P(ND2,ND2)
C                LP(ND2,6),PL(ND2,6)
C                IA(ND4,ND2),AA(ND4,ND2)
C                JSTA(ND3),LOCY(ND3),NAMEY(ND3,5),LWBANY(ND3),LOCY(ND3),
C                     NAMEX(ND3,5),LWBANX(ND3),VRBL(ND3),XDATA(ND3)
C                ECSP(ND5,ND6)
C          ALSO, THE DEFINE FILE STATEMENT SHOULD BE
C                DEFINE FILE 4(NEQ-ND6,ND5,L,K)
      DIMENSION NDATE(728)
      DIMENSION KXTAU( 49),NCAT( 49),CAT( 49),AVG( 49),
     1          SIG( 49),S( 49),AATMP( 49)
      DIMENSION NPRED(2, 49)
      DIMENSION NWORD(4, 49)
      DIMENSION NGP( 20)
      DIMENSION P( 49, 49)
      DIMENSION LP( 49,6),PL( 49,6)
      DIMENSION IA( 20, 49),AA( 20, 49)
      DIMENSION JSTA(260),LOCY(260),NAMEY(260,5),LWBANY(260),LOCX(260),
     1          NAMEX(260,5),LWBANX(260),VRBL(260),XDATA(260)
      DIMENSION ECSP(1275,20)
      EQUIVALENCE (S(1),SIG(1)),(AA(1,1),IA(1,1)),(PL(1,1),LP(1,1)),
     1            (VRBL(1),XDATA(1))
      REAL*8 AVG,SIG,S,P,ECSP
      DATA ND1/728/
      DATA ND2/ 49/
      DATA ND3/260/
      DATA ND4/ 20/
      DATA ND5/1275/
      DATA ND6/20/
C           ABOVE CONSTANTS MUST CORRESPOND TO ACTUAL DECLARED DIMENSIONS
      DEFINE FILE 4(50,1275,L,K)
      CALL M600(NDATE,NPRED,KXTAU,NCAT,CAT,AVG,SIG,S,AATMP,NWORD,LP,PL,
     1          P,IA,AA,NGP,ECSP,JSTA,LOCY,NAMEY,LWBANY,LOCX,NAMEX,
     2          LWBANX,VRBL,XDATA,ND1,ND2,ND3,ND4,ND5,ND6)
      STOP
      END
```

JSTA(ND3), LOCY(ND3), NAMEY(ND3), LWBANY(ND3), LOCX(ND3), NAMEX(ND3), LWBANX(ND3), VRBL(ND3), XDATA(ND3)

ECSP(ND5,ND6)

As the listing of DRM600 shows, certain variables are equivalenced and declared REAL*8. A DEFINE FILE statement must be provided:

DEFINE FILE 4(MAX,ND5,L,K)

where MAX $\geq$ NEQ-ND6 (50 in example above).

CARD INPUT

Card order must be followed as given. All integers are right adjusted.

Card Type 1 - Format (20A4)

This information is printed to identify output.

Card Type 2 - Format (3I4,F4.4,I4,2XZ6,Z4,I4,2F8.0,4I4)

KCYCLE - 00 indicates model run time of 0000 GMT;
12 indicates model run time of 1200 GMT.

NST - Maximum number of predictors to select.
Maximum of 20.

NSTAG - Designates type of equation to be derived (see Card Type 5):
0 = single station equations are derived for all stations.
$\geq$ 1 = generalized operator equations are derived for each of NSTAG groups.

CUTOFF - Screening is continued until NST variables have been selected or until the fraction of the variance of each of the predictand categories explained by the next variable to be selected is less then CUTOFF.

MFORCE - Number of predictors to force before screening.

NSTRAT(1), NSTRAT(2) - together hold the variable on which stratification is to be done, in same format as NPRED(1,) and NPRED(2,). Use 7000000000 for predictand. Zero = no stratification. Stratification on a predictor can be done if it comes from either the predictor or predictand tape.

JSTRAT - Time of stratification variable NSTRAT( ), if any. Not used if NSTRAT( ) = 0 or 7000000000.

STRATL - Lower limit of NSTRAT( ) for stratification. This value included in sample.

-99999 and +99999, respectively.

When NCTYP( ) ≠ 0: For k=1, NCAT1-1, binary predictand k is given the value of 1 if the original variable has a value ≤ CAT1(k); otherwise, it is given the value of 0. For ease of programming, binary predictand NCAT1 is set=1-binary predictand NCAT1-1. That is, the last two categories add to unity.

NCTYP( ) - =0 if exclusive predictand categories are desired; otherwise, cumulative categories will be made (see CAT1( ) above). NCTYP( ) is not used if NCAT1( ) = 0.

Card Type 5 - Format (12I6)

JSTA( ) - One or more groups of numbers are read which define the stations to be used in the regression analysis. There are 2 possibilities depending on the value of NSTAG (Card Type 2, word 3): if NSTAG = 0, one group will be read and single station equations developed for each station; if NSTAG ≥ 1, NSTAG groups will be read and one generalized operator equation developed for each group. A group terminator is 999999; zeros or blanks can exist anywhere in the list.

Card Type 6 - Format (10A8)

TAPS(1,) - List of predictor tape reel numbers to be used on logical unit number 1. A blank word is the terminator. The maximum number of tapes is 8. This list is for printout only; actual tape selection is made from JCL DD cards.

Card Type 7 - Format (10A8)

TAPS(2,) - List of predictand tape reel numbers to be used on logical unit number 2. A blank word is the terminator. The maximum number of tapes is 8. This list is for printout only; actual tape selection is made from JCL DD cards.

Card Type 8 - Format (9I8)

NDATE( ) - Input dates are read 9 per card. These are of the form: 2 digits of the year x 10000 + the number of the month x 100 + the day of the month. These are used to select the data from the predictor and predictand tapes. They must be in ascending order. Blank or zero words may be left on the cards for deleted dates. There must be a 999999 terminator.

NCATF( ) - The category of the predictor. If the predictor was not categorized, this must be zero. If it was categorized, a 1 here would refer to the first category (or binary predictor), etc.

Card Type 11 - Format (20I1)

NPUNCH( )- NST values are read. The value of NPUNCH(k) should be 1 or 0 denoting whether the equation(s) with k predictors is or is not to be punched, respectively.

Card Type 12 - Format (8I1)

JPUNCH( )- As many values are read as these are predictands, including categories, maximum of 8. JPUNCH(k) should be a 1 or 0 signifying whether the equation(s) for the kth predictand is or is not to be punched, respectively.

Card Type 13 - Format (20I1)

NWRITE( )- NST values are read. The value of NWRITE(k) should be 1 or 0 denoting whether the equation(s) with k predictors is or is not to be written on tape 8, respectively.

Card Type 14 - Format (8I1)

JWRITE( )- As many values are read as there are predictands, including categories, maximum of 8. JWRITE(k) should be a 1 or 0 signifying whether the equation(s) for the kth predictand is or is not to be written on tape 8, respectively.

TAPE INPUT

A. Predictor Tape - Data Set Reference Number 1

This is a 9-track, 1600 BPI, binary mode tape read with subroutine RDX. The format of the tape is given in "Format of MOS Interpolated Predictor Tapes." (See example JCL for DD statement.)

B. Predictand Tape - Data Set Reference Number 2

This is a 9-track, 1600 BPI, binary mode tape read with subroutine RDY1. The format of the tape is given in "Format of MOS Predictand Tapes." (See example JCL for DD statement.)

Card Type 5 - Format (12I6)

Four sets of station numbers are read in, each with a 999999 terminator. The four sets consist of 20, 1, 2, and 1 stations.

```
 03812 94224 13876 24011 14739 13880 14819 23062 03927 24143 12960 03947
 23174 12839 94789 23183 24127 23234 13743 24243      999999
 03812      999999
 94224 03812999999
 13876      999999
```

Card Type 6 - Format (8A8)

One predictor tape, E10369, is to be used.

```
E10369
```

Card Type 7 - Format (8A8)

One predictand tape, E10370, is to be used.

```
E10370
```

Card Type 8 - Format (9I8)

16 dates are to be used.

```
 700401  700402  700403  700404  700405  700406  700408  700409  700410
 700411  700412  700413  700414  700415  700416  700418  999999
```

Card Type 9 - Format (Z6,Z4,I4,I2,7F8.0)

Three variables are to be used. The first, the 18-hr unmodified boundary layer V wind component forecast by the PE model (see "PE Variable

OUTPUT: Printout of complete regression analysis, summary statistics, and/or diagnostic information is mostly self explanatory. Regression equations, if punched or written to tape, will be in the format used in the MOS system.

RESTRICTIONS: LABELED COMMON block BLOCK4 is used. Other restrictions are noted above.

NONSYSTEM ROUTINES CALLED: RDX, RDY1, RDY2, RDY3, RDXY, ORDERP, CHNGDT, SUMRY, THROUT, XFER, XPROD, YCMPUT, YXCMPT, SAVREC, YUNPKR, RDLIST, RDAUX, WRAUX, AND W3AI01.

STORAGE REQUIREMENTS: Depends on dimensions furnished by DRM600. The example DRM600 shown requires 384K bytes.

LANGUAGE: Fortran IV (H Extended).

LOCATION: M600 is on load module DSN=NWS.SDO.TDL.LOAD with member name OPM600. All subroutines are provided on load modules (see example JCL) except YCMPUT and YXCMPT, which are for special applications and must be provided by the user only if needed. See Card Types 4 and 9 for explanations of use of YCMPUT and YXCMPT. W3AI01 is on load module DSN=NWS.NMC.TEST.LOAD with member name W3AI01. Others are on load module DSN=NWS.SDO.TDL. LOAD with member names the same as subroutine names with a prefix of "OP". For instance RDX has member name OPRDX.

MOS OP NO. 11

RDLIST

READS A SET OF DATA CARDS

George W. Hollenbaugh
Harry R. Glahn
October 20, 1974

PURPOSE: This routine reads from cards, according to a given format, a set of data for use by a calling program. The cards are read one at a time into a temporary array and each word checked for the terminator before saving the element in a permanent named array. Blank or zero words may be left on the cards for deleted elements which will be ignored if the terminator is not zero or blank. An extra card will be necessary for the terminator if the last card of the card set is filled.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
DIMENSION JV( ), MP( )
.
CALL RDLIST (JV,NV,MP,NP,MA,NN,MT)
.
```

JV - Array in which data will be returned.

NV - Size of array JV.

MP - Temporary array.

NP - Size of temporary array MP. Must equal the number of words per card.

MA - Contains format of data--maximum of 8 characters.

NN - Count of elements returned in array JV.

MT - Terminator.

EXAMPLE:

```
DIMENSION IDATES(365),ITEMP(9)
CALL RDLIST(IDATES,365,ITEMP,9,8H(9I8)    ,NCAS,999999)
```

Cards are read one at a time, 9 items per card, with FORMAT (9I8) until the terminator 999999 is reached. Blanks and zeros are omitted and the data, sans terminator, are placed into IDATES( ). A maximum

MOS OP NO. 12

M505

CHECKS MOS ASHEVILLE DATA

George W. Hollenbaugh
January 22, 1975

PURPOSE: The National Climatic Center compiles a monthly summary of weather observations for selected stations and puts these reports on magnetic tape in a format requested by TDL. The purpose of this program is to check the observations for each station by means of a combination automatic-manual procedure and put the data into a form described in "Format of MOS Predictand Tapes."

CARD INPUT:

Card order must be followed as given. All integers are right adjusted.

Card Type 1 - Format (5I4)

NDAY - Total number of days on the tape to be processed.

NOUT - Non-zero if a binary packed data tape is to be written.

MATX - Non-zero if a 2-dimensional matrix of stations and parameters is to be printed. The count is the number of times the parameter was reported as missing.

NSTANS - Total number of stations to be used.

LSTRCD - Non-zero if each processed report is to be printed.

Card Type 2 - Format (4X,I5,30X,5A4,1X,I4)

NSTANS cards are read here, each of the form:

LSTANS( ) - 5-digit climatological (WBAN) number of station to be used.

NAMEY(,J),J=1,5 - 20 character description (station name) of LSTANS( ).

LOCSTA( ) - Station area code, where: 1=conterminuous U.S., 2=Alaska, 3=Hawaii, and 4=Puerto Rico.

Card Type 3 - Format (I4,4X,I8)

NDAY cards, followed by appropriate cards of Type 4, are read here,

Card Type 2 - Format (4X,I5,30X,5A4,1X,I4)

```
FLG 03103                              FLAGSTAFF, ARIZ       1
```

255 station directory cards are read but only the first is shown. This list will be printed for your convenience.

Card Type 3 - Format (I4,4X,I8)

```
   3    74100103
```

3 adjustments are to be made to the data for date 741001. Thirty-one of these adjustment and date-time cards are read but only the first is shown.

Card Type 4 - Format (53A1,11X,2I4)

```
03103741000106---5150000000000004205008064000003902900O            1   1
```

The station report of Flagstaff, Ariz. (WBAN No. 03103) for date-time 74100106 is to be corrected. Its position in the list is first.

```
0381374100115                                                      4   2
```

The station report of Macon, Ga. (WBAN No. 03813) for date-time 74100115 is to be deleted. It turns out that duplicate reports exist for the station and date-time, and the one out of position must be ignored. The position of the duplicate report is fourth. Note that only the station number and date-time need be punched for the report.

```
038567410012499999999999999999999999999999999999999999            6   3
```

The station report of Huntsville, Ala. (WBAN No. 03856) for date-time 74100124 is to be inserted. The station was mistakenly omitted from the list. Its position is sixth. Since no information is readily available, 9's are punched on the card for missing data.

EXAMPLE JCL:

```
//WBM505   JOB (WE20008C294042C,GRAMX),HOLLENBAUGH,CLASS=C,REGION=150K
//        EXEC PROC=NFORXLG
//LKED.ARLIB DD DSN=NWS.SDO.TDL.LOAD,DISP=SHR
//LKED.SYSIN DD *
 INCLUDE ARLIB(OPM505,OPRDASH,OPWRTKT,OPYPKR)
/*
 ENTRY MAIN
//GO.FT01F001  DD DSN=ADATA,VOL=SER=PW025,UNIT=TAPE7,LABEL=(,BLP,,IN),
//             DCB=(DEN=1,RECFM=FB,LRECL=53,BLKSIZE=3392,TRTCH=ET),
//             DISP=(OLD,KEEP)
//GO.FT02F001  DD DSN=YDATA,VOL=SER=E11797,UNIT=TAPE9,LABEL=(,SL,,OUT),
//             DCB=(DEN=3,RECFM=VS,LRECL=5644,BLKSIZE=5648),
//             DISP=(NEW,KEEP)
//GO.SYSIN     DD *
 (INPUT CARDS)
/*
//
```

MOS OP NO. 13

M510

STACKS MOS PREDICTAND DATA

George W. Hollenbaugh
November 8, 1974

PURPOSE: The predictand data, after being error-checked, for any given month, resides on a magnetic tape in packed form. This routine reads each of the data tapes for several months, and in turn, stacks the records onto one tape.

CARD INPUT:

Card order must be followed as given. All integers are right adjusted.

Card Type 1 - Format (I4,17I4)

NTAPS - Number of input tapes to be used--maximum of 17.

IFLAG(J),J=1,NTAPS - Indicates what is written on the output tape when the end of input tape J is reached. 0 = no dummy record written when end of input tape is reached. 1 = dummy record written. If another input tape is used, the dummy record will be followed by header records. If there are no more input tapes, an EOF is written.

TAPE INPUT: Predictand tape - Data Set Reference Number 1

9-track, 1600 bpi, binary mode tapes are read with the asychronous FORTRAN READ statement. Each tape contains one month of packed data preceeded by seven header records. See "Format of MOS Predictand Tapes."

TAPE OUTPUT: Stacked Predictand tape - Data Set Reference Number 2

A 9-track, 1600 bpi, binary mode tape is written with the asychronous FORTRAN WRITE statement. Each tape contains only one file of data (one data set); however it may contain more than one "pseudo file." See "Format of MOS Predictand Tapes."

PRINTOUT: The printout will contain selected information read from data cards described above. Also a list of dates will be printed for which data are saved.

EXAMPLE JCL:

```
//WBM510   JOB (WE20008C294042C,GRAMX),HOLLENBAUGH,CLASS=C,REGION=155K
//         EXEC PROC=NFORXLG
//LKED.ARLIB DD DSN=NWS.SDO.TDL.LOAD,DISP=SHR
//LKED.SYSIN DD *
 INCLUDE ARLIB(OPM510,OPCOPYI,OPCOPYH,OPCOPYD,OPPATCH)
/*
 ENTRY MAIN
//GO.FT01F001  DD DSN=YDATA,VOL=SER=E11660,UNIT=TAPE9,LABEL=(,SL,,IN),
//             DISP=(OLD,KEEP)
//GO.FT01F002  DD DSN=YDATA,VOL=SER=E11661,UNIT=TAPE9,LABEL=(,SL,,IN),
//             DISP=(OLD,KEEP)
//GO.FT02F001  DD DSN=YDATA,VOL=SER=E11662,UNIT=TAPE9,LABEL=(,SL,,OUT),
//             DCB=(DEN=3,RECFM=VS,LRECL=5644,BLKSIZE=5648),
//             DISP=(NEW,KEEP)
//GO.SYSIN     DD *
 (INPUT CARD)
/*
//
```

MOS OP NO. 14

M515

SORTS MOS PREDICTAND TAPES

George W. Hollenbaugh
November 11, 1974

PURPOSE: To stratify by hours and season data records taken from MOS system predictand tapes for use with MOS system programs.

CARD INPUT:

Card order must be followed as given. All integers are right adjusted.

Card Type 1 - Format (I4,17I4)

NTAPS - Number of input tapes to be used--maximum of 17.

ITAG(J),J=1,NTAPS - Indicates what is written on the output tape when the end of input tape J is reached. 0 = no dummy record written when end of input tape is reached. 1 = dummy record written. If another input tape is used, the dummy record will be followed by header records. If there are no more input tapes, an EOF is written.

Card Type 2 - Format (2I4,2(2XI4),2(2XI6))

IOUR1 - Predictand data are to be saved on tape for hour IOUR1.

IOUR2 - Predictand data are to be saved on tape for hour IOUR2.

IBMAD - Beginning month and day in the form month X 100 + day of seasonal data to be saved.

IEMAD - Ending month and day in the form month X 100 + day of seasonal data to be saved.

IBDAT - Beginning date in the form year X 10000 + month X 100 + day which to start saving the data.

IEDAT - Ending date in the form year X 10000 + month X 100 + day to be the last record of saved data.

COMMENTS AND RESTRICTIONS:

IOUR1 and IOUR2 cannot both be zero. Note from "Format of MOS Predictand Tapes" that 0000 GMT for, say, November 1 is identified as hour 24 on October 31.

If only one hour of sorted data is desired, say hour 12, it is immaterial if IOUR1 be left blank and IOUR2 = 12 or IOUR1 = 12 and IOUR2 is left blank.

If two hours of sorted data are desired, say 6 and 18, it is immaterial if IOUR1 = 18 and IOUR2 = 6 or IOUR1 = 6 and IOUR2 = 18. The data for hour 6 will be followed by the data for hour 18 for each day.

If IOUR1 and IOUR2 are equal, say hour 24, the data are saved only once.

IBMAD and IEMAD can be interpreted as describing seasonal data "from IBMAD = 0401 thru IEMAD = 0930" or "from IBMAD = 1001 thru IEMAD = 0331" or yearly data "from IBMAD = 0101 thru 1231."

No more than 17 predictand tapes can be read per run.

Only 1 sorted predictand tape can be written per run.

PEATMOS data exists for hours 6, 12, 18, and 24 GMT. MOS data exists for hours 3, 6, 9, 12, 15, 18, 21, and 24 GMT. Therefore, do not expect to obtain PEATMOS sorted data for hours 3, 9, 15, and 21 GMT because it is not available.

NONSYSTEM ROUTINES CALLED: RDJ, WTY and PATCH.

STORAGE REQUIREMENTS: 200K is sufficient.

LANGUAGE: FORTRAN IV (H Extended Plus).

LOCATION: Exists on load module DSN=NWS.SDO.TDL.LOAD with member name OPM515.

MOS OP NO. 15

M630

MOS BIAS PROGRAM

George W. Hollenbaugh
Harry R. Glahn
November 20, 1974

PURPOSE: To compute the conditional relative frequency of specified categories of a variable taken from a MOS predictand tape as a function of categories of a variable taken from a MOS predictor tape.

CARD INPUT:

Card order must be followed as given. All integers are right adjusted.

Card Type 1 - Format (3I4,2F8.0,2XA2)

MYTAU - Predictand projection time in hours. Actual predictand times are NDATE( ) + MYTAU.

NPOS - Column within the predictand matrix where the observations reside that are to be matched with the forecasts. See "Predictand Data Matrix Description."

KCYCLE - 00 indicates model run time of 0000 GMT.
12 indicates model run time of 1200 GMT.

CAT(J),J=1,2 - Limits for categorizing the predictand. The number of limits to be read must be 2.

NGO - LE (or GE) for relative frequencies equal to or less than (equal to or greater than) CAT( ).

Card Type 2 - Format (12I6)

JSTA( ) - 5-digit climatological (WBAN) numbers of stations to be used in the analysis. Blank words may be left on the cards for deletions. The terminator is a 999999 word.

Card Type 3 - Format (9I8)

NDATE( ) - Dates in ascending order. They must appear in the form year X 10000 + month X 100 + day. Blank words may be left on the cards for deletions. The terminator is a 999999 word.

TAPE INPUT:

A. Predictor tape - Data Set Reference Number 1

9-track, 1600 bpi, binary mode tape, read with subroutine RDX. Format of the tape is given in "Format of MOS Interpolated Predictor Tapes." (See example JCL for DD statement.)

B. Predictand tape - Data Set Reference Number 2

9-track, 1600 bpi, binary mode tape, read with subroutine RDY1. Format of the tape is given in "Format of MOS Predictand Tapes." (See example JCL for DD statement.)

PRINTOUT:

The primary output will consist of the relative frequency of each of 2 predictand categories for each of 3 predictor categories for each station in JSTA( ). These relative frequencies will be listed by station and can be plotted on maps if desired (see IMAPS( )). Also, selected information read from the data cards will be printed.

EXAMPLE INPUT CARDS:

Card Type 1 - Format (3I4,2F8.0,2XA2)

The predictand (12-hr precipitation amount) is in column 15 of the data matrix; a projection of 24 hours is desired from the run time of 0000 GMT. The two predictand categories are defined such that the first (second) category will be composed of those cases having a 12-24 hour precipitation amount greater than or equal to .009 (.5) inches.

Card Type 2 - Format (12I6)

These are the numbers that define the stations to be used. The terminator is 999999. Zero or blank words can exist anywhere in the list.

Card Type 3 - Format (9I8)

These are the eight dates for which the conditional relative frequency will be computed. The terminator is 999999. Zero or blank words can exist anywhere in the list.

COMMENTS AND RESTRICTIONS:

NPOS must be in the range of 1 thru 18.

No more than 9 predictor tapes can be used at any one time.

No more than 9 predictand tapes can be used at any one time.

The maximum number of JSTA's is 260.

The maximum number of NDATE's is 1008.

NONSYSTEM ROUTINES CALLED: RDY1, RDY2, RDY3, RDXY, CHNGDT, YUNPKR, RDX, RDLIST, MAPS and W3AI01.

STORAGE REQUIREMENTS: 225K is sufficient.

LANGUAGE: FORTRAN IV (H Extended Plus).

LOCATION: M630 exists on load module DSN=NWS.SDO.TDL.LOAD with member name OPM630; all subroutines except W3AI01 are on the same module with member names the same as the subroutine names with a prefix of "OP". W3AI01 is on the load module DSN=NWS.NMC.TEST.LOAD (see example JCL).

MOS OP NO. 16

MAPS

PLOTS NUMERIC VALUES ON A U.S. MAP

Frederick Marshall
November 18, 1974

PURPOSE: This subroutine is designed to plot single numeric values in the range -999 to 999 at any location in the conterminous U.S., southern Canada, and northern Mexico. Locations can be specified by latitude and longitude, x and y coordinates, or WBAN numbers.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
DIMENSION FLAT( ), FLON( ), IDATA( )
.
.
CALL MAPS (NSTA,FLAT,FLON,IDATA,LOC,NBK)
.
```

NSTA - Number of stations for which values are to be plotted. (INTEGER*4)

FLAT( ) - Array containing WBAN numbers (INTEGER*4), latitudes (REAL*4), or x coordinates (REAL*4), depending, respectively, on whether LOC = 0, 1, or 2 (see LOC below).

FLON( ) - Array contining dummy values, longitudes (REAL*4), or y coordinates (REAL*4), depending, respectively, on whether LOC = 0, 1, or 2. Order of values in FLON( ) must correspond to order in FLAT( ).

IDATA( ) - Array containing numeric values to be plotted. (INTEGER*4)

LOC - Type of station location used, where: 0 = WBAN numbers, 1 = latitude and longitude in degrees and hundredths, and 2 = x and y coordinates in terms of the grid where (1,1) respectively is the lower left hand corner of the grid.

NBK - Map background desired, where: 0 = none, 1 = U.S. outline only, and 2 = U.S. and state outlines.

EXAMPLE 1 :

```
DIMENSION FLAT(267),FLON(267),IDATA(267)
CALL MAPS (267,FLAT,FLON,IDATA,1,1)
```

M300

MERGES TWO MOS INTERPOLATED PREDICTOR TAPES

Frank T. Globokar
Sept. 1, 1974
December 31, 1974 (Rev.)

PURPOSE: Subroutine merges two files of MOS interpolated predictor data onto a single file. Dates on each of the input files, TAPE1 and TAPE2, must be ordered. Output on file TAPE3 will also be ordered by date. All data are in the format described in "Format of MOS Interpolated Predictor Tapes." Typically, one input file will have been prepared by program M200 and will contain data from one model. The other input file will have been prepared by M200 and will contain data from a different model. M300 uses variable dimensions so that storage requirements can be altered as needed. A driver DRM300 is available in source form.

CALL AND EXPLANATION OF DRIVER DRM300

A list of DRM300 follows:

```
C     PROGRAM DRM300
C        PURPOSE
C        DRIVER FOR M300 MERGING SUBROUTINE
C     LET ND1 = NUMBER GREATER OR EQUAL TO THE NUMBER OF STATIONS
C               USED IN M200 INTERPOLATION PROGRAM.
C         ND2 = NUMBER GREATER OR EQUAL TO THE SUM OF PREDICTORS ON
C               TAPES TO BE MERGED
C     DIMENSION NAMEX(ND1,5),LWBANX(ND1),LWBANY(ND1),IDATA(ND1)
C     DIMENSION IPRED(ND2,7),KPRED(ND2,7),NPRED(ND2,7)
C*****USING DISCRIPTION ABOVE, THE USER MUST SUPPLY NEXT 4 CARDS.
      DIMENSION NAMEX(260,5),LWBANX(260),LWBANY(260),IDATA(260)
      DIMENSION IPRED(400,7),KPRED(400,7),NPRED(400,7)
      DATA ND1/260/
      DATA ND2/400/
      CALL M300(NAMEX,LWBANX,LWBANY,IDATA,IPRED,KPRED,NPRED,ND1,ND2)
      STOP
      END
```

In DRM300, 2 variables are given values in DATA statements. These are:

ND1 - Set ≥the number of stations used in both runs of M200 interpolation programs. For consistency among MOS programs, use ND1 = 260 unless a larger number is actually needed.

ND2 - Set ≥ the total number of predictors on the two tapes to be merged.

TAPE OUTPUT: Data Set Reference Number 3

This is also a 9-track, 1600 bpi, binary mode tape written with the ASYNCHRONOUS WRITE statement. The order of fields on tape 3 is by date. All fields (for a particular date) from tape 1 will appear first, followed by all fields on tape 2. Also, for a particular date, the order of fields on tapes 1 and 2 will be maintained on tape 3. Only matching dates will be saved on tape 3. The cycle time on the output tape will correspond to the cycle time of the model on tape 1. This feature is primarily for merging 06Z (18Z) SUM runs with 00Z (12Z) PE and Trajectory runs. The 06Z SUM will become 00Z, but the projections (taus) will not be modified. That is, a SUM tau of 6 will verify at 12Z, while a PE tau of 6 will verify at 06Z. This procedure is necessary for certain analysis programs, such as M600, to run properly.

EXAMPLE INPUT CARDS:

Card Type 1 - Format (20A4)

```
TEST OF TJ AND LFM MERGE
```

Card Type 2 - Format (2I4)

```
00  0
```

The cycle time of both models is the same and the matching dates will be punched.

Card Type 3 - Format (9A8)

```
E11111  E12222
```

Two input tapes, E11111 and E12222, are to be used on Data Set Reference Number 1.

Card Type 4 - Format (9A8)

```
E15555  E16666
```

Two input tapes E15555 and E16666, are to be ueed on Data Set Reference Number 2.

Card Type 5 - Format (9A8)

```
E11212  E11313  E11414  E11515
```

Four output tapes, E11212, E11313, E11414 and E11515, are provided for output on Data Set Reference Number 3.

EXAMPLE JCL:

```
//MGM300 JOB (0E20008C2960410,GRAFX),'GLOBOKAR TOL',
//  REGION=200K,TIME=5,MSGLEVEL=(1,1)
//             EXEC WFQ8XCLG,PARM.FORT=(XL)
//LKED.MYLIB DD DSN=MWS.SOL.TOL.LOAD,DISP=SHR
//LKED.SYSIN DD *
  INCLUDE MYLIB(DRM300)
  ENTRY MAIN
/*
//FORT.SYSIN DD *

          (DRIVER DRM300)

/*
//GO.FT01F001 DD DSN=XDATA,DISP=(OLD,KEEP,KEEP),UNIT=TAPE9,
//   LABEL=(1,SL,,IN),VOL=SER=F11111,
//   DCB=(RECFM=VS,LRECL=8196,BLKSIZE=8200,DEN=3)
//GO.FT01F002 DD DSN=XDATA,DISP=(OLD,KEEP,KEEP),UNIT=TAPE9,
//   LABEL=(1,SL,,IN),VOL=SER=F12222,
//   DCB=(RECFM=VS,LRECL=8196,BLKSIZE=8200,DEN=3)
//GO.FT02F001 DD DSN=XDATA,DISP=(OLD,KEEP,KEEP),UNIT=TAPE9,
//   LABEL=(1,SL,,IN),VOL=SER=F15555,
//   DCB=(RECFM=VS,LRECL=8196,BLKSIZE=8200,DEN=3)
//GO.FT02F002 DD DSN=XDATA,DISP=(OLD,KEEP,KEEP),UNIT=TAPE9,
//   LABEL=(1,SL,,IN),VOL=SER=F16666,
//   DCB=(RECFM=VS,LRECL=8196,BLKSIZE=8200,DEN=3)
//GO.FT03F001 DD DSN=XDATA,DISP=(NEW,KEEP,KEEP),UNIT=TAPE9,
//   LABEL=(1,SL,,OUT),VOL=SER=F11212,
//   DCB=(RECFM=VS,LRECL=8196,BLKSIZE=8200,DEN=3)
//GO.FT03F002 DD DSN=XDATA,DISP=(NEW,KEEP,KEEP),UNIT=TAPE9,
//   LABEL=(1,SL,,OUT),VOL=SER=F11313,
//   DCB=(RECFM=VS,LRECL=8196,BLKSIZE=8200,DEN=3)
//GO.FT03F003 DD DSN=XDATA,DISP=(NEW,KEEP,KEEP),UNIT=TAPE9,
//   LABEL=(1,SL,,OUT),VOL=SER=F11414,
//   DCB=(RECFM=VS,LRECL=8196,BLKSIZE=8200,DEN=3)
//GO.FT03F004 DD DSN=XDATA,DISP=(NEW,KEEP,KEEP),UNIT=TAPE9,
//   LABEL=(1,SL,,OUT),VOL=SER=F11515,
//   DCB=(RECFM=VS,LRECL=8196,BLKSIZE=8200,DEN=3)
//GO.SYSIN DD *

          (DATA CARDS)
/*
//
```

M200PE,M200TJ,M200LM,M200SM,M200BL

INTERPOLATES TO STATIONS FROM MOS GRID-POINT TAPES

Frank T. Globokar
August 1, 1974
December 31, 1974 (Rev.)

PURPOSE: The objectives of the interpolating and abstracting subroutines are to interpolate from grid-point data fields on MOS tapes to designated stations or locations, and to perform, through subroutine OPTION and other user-provided subroutines, computations of new variables. The basic grid tapes contain data from the PE, TJ, LFM, SUM, or PBL models.

Each record from the grid tapes includes a 12-word NMC identifier and a value for each grid point (packed). (See "Format of MOS Grid-Point Predictor Tapes".) The data for a selected day are read from the tape and saved in memory. At this point, the data fields are available for retrieval for computations necessary to computing other fields from the existing fields if that is desired. Interpolation to the designated stations is performed on all the intended fields and values saved on the output tape. M200 utilizes variable dimensions so that storage requirements can be altered as needed. A driver DRM200 is available in source form.

CALL AND EXPLANATION OF DRIVER DRM200

```
C       PROGRAM DRM200
C          PURPOSE
C             DRIVER FOR M200 INTERPOLATION SUBROUTINES
C          THIS DRIVER AND APPROPRIATE OPTION SUBROUTINE ARE
C          SUFFICIENT TO RUN INTERPOLATION PROCESS.
C          SEE M200 WRITEUP FOR COMPLETE EXPLANATION.
C   MD1 = NUMBER GREATER OR EQUAL TO THE NUMBER OF PREDICTORS.
C             FOR PE MODEL MD1 SHOULD BE GREATER OR EQUAL TO (2 TIMES
C             OF PREDICTORS PLUS 6.
C       MD2 = 176 FOR PE MODEL INTERPOLATION.
C             124 FOR TJ MODEL INTERPOLATION.
C             304 FOR PBL OR LFM INTERPOLATION.
C             1272 FOR SUM INTERPOLATION.
C       MD3 = NUMBER GREATER OR EQUAL TO THE NUMBER OF POINTS OR STATIONS
C             YOU ARE INTERPOLATING TO.
C       MD4 = MD3+3
C       DIMENSION ISAV(MD2,201)
C       DIMENSION AA(MD4)
C       DIMENSION BB(MD3,6),CC(MD3),DD(MD3),EE(MD3),FF(MD3),GG(MD3),
C                 HH(MD3),II(MD3),JJ(MD3),KK(MD3),LL(MD3),MM(MD3),
C                 NN(MD3),OO(MD3),PP(MD3),QQ(MD3),RR(MD3),SS(MD3),
C                 TT(MD3),UU(MD3),VV(MD3),WW(MD3),XX(MD3),YY(MD3)
C****** THE NEXT 6 CARDS SHOULD BE INSERTED WITH THE PROPER DIMENSIONS
      DIMENSION ISAV(224,50)
      DIMENSION AA(263)
      DIMENSION BB(260,6),CC(260),DD(260),EE(260),FF(260),
     1          GG(260),HH(260),II(260),JJ(260),KK(260),LL(260),MM(260),
     2          NN(260),OO(260),PP(260),QQ(260),RR(260),SS(260),TT(260),
     3          UU(260),VV(260),WW(260),XX(260),YY(260)
C       THE FOLLOWING DIMENSION STATEMENT DOES NOT CHANGE.
      DIMENSION DATAIRE(20),ITEST(3)
      EQUIVALENCE(AA(1),ITEST(1))
      EQUIVALENCE (PP(1),BB(1)),(RR(1),SS(1)),(TT(1),DD(1)),
     1            (VV(1),CC(1)),(XX(1),YY(1))
      COMMON/BLOCK1/MRGDZ,ITAPE,KTAPE,KSKIP
C****** DATA STATEMENTS FOR MD1,MD3,MD4 MUST BE INSERTED
      DATA MD1/176/
      DATA MD3/260/
      DATA MD4/263/
      READ(5,100)(DATAIRE(K),K=1,20)
  100 FORMAT(20A4)
```

In DRM200, 3 variables are given values in DATA statements. These are:

ND1 - Set ≥ the number of predictor fields identified in Card Type 4 described in "Card Input" below; for the PE model ND1 should be ≥ the number of predictor fields +4 (90 in example above).

ND3 - Set ≥ the number of stations (or points) in card type 8 for which interpolation is done (260 in example above.)

ND4 - Set ND3 + 3 (263 in example above.)

Also in DRM200, 27 variables are given constant dimensions. These dimensions must agree with the values of the variables ND1, ND3, and ND4 as shown below:

ISAV (ND2,ND1)

AA (ND4)

BB (ND3,4)

CC (ND3), DD(ND3), EE(ND3), FF(ND3), GG(ND3), HH(ND3),II(ND3), JJ(ND3), KK(ND3), LL(ND3), MM(ND3), NN(ND3), OO(ND3), PP(ND3), QQ(ND3), RR(ND3), SS(ND3), TT(ND3), UU(ND3), VV(ND3), WW(ND3), XX(ND3), YY(ND3).

The first dimension of ISAV(,), ND2, must be given a value as indicated in the table below:

Model Being Used	Subroutine called	ND2	IMOD(card type 2)
PE	M200PE	176	0
Trajectory	M200TJ	123	1
LFM	M200LM	384	2
SUM	M200SM	1272	3
PBL	M200BL	384	4

Statement 107 in DRM200 must call the appropriate subroutine as indicated in the table above.

As the listing of DRM200 shows, certain variables are equivalenced and used in common block BLOCK1. The 9 input card types required are described below. The first two are read by DRM200 and the last 7 by the subroutine being used.

CARD INPUT:

Card order must be followed as given. All integers are right adjusted.

Card Type 1 - Format (20A4)

NATURE( ) - This information is printed to identify output (one card).

5 = This field is a computed field not available on basic grid tapes; it is used to compute additional computed fields, Interpolation of original computed field not saved.

6 = Signifies values are to be read from cards for specific stations (see card type 9). An example would be station elevations.

LABEL(,5), LABEL(,6), LABEL(,7), LABEL(,8) - 15-character field used to identify data. Not needed for basic fields (i.e. if processing indicator LABEL (,4) is a 1, 3, or if field is a smoothed basic field).

Fields may be punched 1 or 2 per card. A completely blank card acts as a terminator and signifies the end of card type 4. The order of fields in card type 4 is the order that they will appear on the output tape. This order need not be the same as the order on the input tapes.

Card Type 5 - Format (70I1)

NB( ) - Map output designators, one for each field identified in card type 4 and in that same order. A non-zero entry produces a contoured grid map for each date in IDATE( ), unless predictor is a computed field for which no scaling and contouring constants exist in the program; in the latter case, see Frank Globokar.

Card Type 6 - Format (9A8)

ITAPS( ) - List of grid-point input tape reel numbers to be used on data set reference number 1. A blank word is the terminator. The maximum number of tapes is 36. A blank card is necessary if the list of tapes terminates in columns 65-72 except when exactly 36 are used, no blank card is necessary. This list is for printout only; actual tape selection is made from JCL DD cards.

Card Type 7 - Format (9A8)

KTAPS( ) - List of output tape reel numbers to be used on data set reference number 2. A blank word is the terminator. The maximum number of tapes is 18. A blank card is necessary if the list of tapes terminates in column 65-72. This list is for printout only; actual tape selection is made from JCL DD cards.

TAPE INPUT - Data Set Reference Number 1

Basic grid-point tape. This is a 9 track, 1600 BPI, binary mode tape read with the FORTRAN READ statement. Format of tape is given in "Format of MOS Grid-Point Predictor Tapes."

TAPE OUTPUT - Data Set Reference Number 2

Interpolated station data. This is a 9 track, 1600 BPI, binary mode tape written with the FORTRAN WRITE statement. Format of the tape is given in "Format of MOS Interpolated Predictor Tapes."

EXAMPLE INPUT CARDS

Card Type 1 - Format (20A4)

These 70 characters identify the output.

```
      TEST OF TRAJECTORY INTERPOLATION PROGRAM
```

Card Type 2 -Format (I8,3I4)

Output tape will be skipped until positioned after day 721030 (October 30, 1972). The run time of the model is 12Z. Interpolation will be done on fields from the trajectory model. Up to 3 days will be skipped because of missing data or parity errors before program halts.

```
  721030  12   1   3
```

Card Type 3 - Format (9(2X,I6)

11 dates are to be used.

```
  721101  721102  721103  721104          721107  721108  721109  721110
  721111  721120  721121  999999
```

Card Type 4 - Format (2(1X,Z6,Z4,I3,1X,I1,1X,4A4,7X))

Fields to be used by program, 1 or 2 to a card. The first field 2310200001 24, with indicator 1, is a basic field and will be used in computation but not saved on tape. Field 2310200101 24, with indicator 2, is a computed field. The eighth digit, a 1, indicates this field is a 5 pt. smoothed field of the 2310200001 24, the first field. Since the first field is a basic field, the program will perform the smoothing without calling a subroutine from OPTION. Field 3333330001 00,

Card Type 8 - Format (R3,1X,I5,2X,F3.0,1X,F2.0,1X,F3.0,1X,F2.0,15X,5A4,I5)

Interpolation will be done for these 7 stations, The 1 in column 64 will be ignored in all cases except for the PE in which case it is referenced to determine which PE grid is being used.

```
FLG 03103   35 08 111 40.                FLAGSTAFF, ARIZ              1
AGS 03820   33 22  81 58                 AUGUSTA,GA                   1
HSV 03856   34 42  86 35                 HUNTSVILLE, ALA              1
GSW 03927   32 50  97 03                 FORT WORTH, TEX              1
FMY 12835   26 35  81 52                 FORT MYERS, FLA              1
HAR 14751   40 13  76 51                 HARRISBURG, PA               1
ACK 14756   41 15  70 04                 NANTUCKET, MASS              1
```

Card Type 9

Card set 1 - Format (Z6,Z4,I3)

The predictor in card type 4 with indicator 6 which has data following in card set 2.

```
3333330001 00
```

Card Set 2 - Format (4(3X,I5,2X,F10.0))

The station numbers and corresponding values at those stations of above predictor.

```
   03820          8.   03103          7.   03856          6.
   14764         10.   12835         11.   14751         13.   03927         10.
```

OUTPUT: A listing of station , fields, and dates for which interpolations are made.

RESTRICTIONS: LABELED COMMON BLOCK1 thru BLOCK9, BLOCKA, and BLOCKB are used.

A driver program DRM200 and the subroutine OPTION will be needed for any interpolation run. The other subroutines (except those for special computations called from OPTION) will be on the load module DSN=NWS.SDO.TDL.LOAD (see example JCL cards).

EXAMPLE JCL AND DECK SETUP:

```
//WGM200 JOB (RE2000BC29404)0,GRAHX),'GLUBOKAR TDL',
//  REGION=400K,TIME=5,MSGLEVEL=(1,1)
//          EXEC NEWPXCLG,PARM.FORT=(XL)
//FORT.SYSIN DD *
```

(DRIVER DRM200 AND OPTION SUBROUTINE INSERTED HERE)

```
/*
//LKED.LIB1  DD DSN=NWS.SDD.TDL.LOAD,DISP=SHR
//LKED.MYLIB DD DSN=NWS.NMC.TEST.LOAD,DISP=SHR
//LKED.SYSIN DD *
  INCLUDE MYLIB(W3AI00,W3AI01,GRDPRT,W3FB00)
  INCLUDE LIB1(OPM200TJ)   (Interpolation subroutine you wish)
  INCLUDE LIB1(OPM200SB)   (Includes necessary subroutines)
  INCLUDE LIB1(OPRCHECK,OPNCHECK,OPRPECS,OPWECS)
  ENTRY MAIN
/*
//GO.FT01F001 DD DSN=TJGRID,DISP=(OLD,KEEP,KEEP),UNIT=TAPE9,
//  LABEL=(1,SL,,IN),VOL=SER=WB924,
//  DCB=(RECFM=VBS,LRECL=492,BLKSIZE=9924,DEN=3)
//GO.FT01F002 DD DSN=TJGRID,DISP=(OLD,KEEP,KEEP),UNIT=TAPE9,
//  LABEL=(1,SL,,IN),VOL=SER=WB926,
//  DCB=(RECFM=VBS,LRECL=492,BLKSIZE=9924,DEN=3)
//GO.FT02F001 DD DSN=XDATA,DISP=(NEW,KEEP,KEEP),UNIT=TAPE9,
//  LABEL=(1,SL,,OUT),VOL=SER=F19499,
//  DCB=(RECFM=VS,LRECL=4196,BLKSIZE=8200,DEN=3)
//GO.SYSIN DD *
```

(DATA CARDS)

```
/*
//
```

EXAMPLE OPTION SUBROUTINES:

(For TJ) - SUBROUTINE OPTION (MH,NIWDS,NRCDZ,ISAV,NECS,IGRD,JJ,II,JDATE)

```
DIMENSION US (13,17),UA(13,17),UT(13,17)
DIMENSION MH (NIWDS),ISAV(NIWDS,NECS)
COMMON/BLOCKA/FD1(224),FE2(224),FD3(224)
EQUIVALENCE (FD1(4),US(1,1)),((FD2(4),UA(1,1)),(FD3(4),UT(1,1))

(Using the example mentioned in card type 4, the following
subroutine could be called).

CALL DIFF(IHX(231020),24,IHX(231021),24,0,0,0,IHX(23102).
JDATE,JH,NIWDS,NRCDZ,ISAV,NECS,US,UA,UT,JJ,II,FD1,FD2,FD3,IGRD)

RETURN
END
```

(For PE) - SUBROUTINE OPTION (MH,NIWDS,NRCDZ,ISAV,NECS,IGRD,JJ,II,JDATE)

```
DIMENSION US(15,17),AK(8,7),WH(4,3),PI(2,2)
DIMENSION  UA15,17),AS(8,7),WI(4,3),PR(2,2)
DIMENSION UT(15,17),AL(8,7),WA(4,3),PT(2,2)
DIMENSION MH(NIWDS),ISAV(NIWDS,NECS)
COMMON/BLOCKA/FD1(330),FD2(330),FD3(330)
EQUIVALENCE (FD1(4),US(1,1)),(FD1(259),AK(1,1)),(FD1(315),WH(1,1)),
   (FD1(327),PI(1,1))
EQUIVALENCE (FD2(4),UA(1,1)),(FD2(259),AS(1,1)),(FD2(315),WI(1,1)),
   (FD2(327),PR(1,1))
EQUIVALENCE (FD3(4),UT(1,1)),(FD3(259),AL(1,1)),(FD3(315),WA(1,1)),
   (FD3(327),PT(1,1))

(Subroutines you wish to call)

RETURN
END
```

(For LFM or PBL) - SUBROUTINE OPTION (MH,NIWDS,NRCDZ,ISAV,NECS,IGRD,JJ,II,JDATE)

```
DIMENSION US(24,31),UA(24,31),UT(24,31)
DIMENSION MH(NIWDS),ISAV(NIWDS,NECS)
COMMON/BLOCKA/FD1(747),FD2(747),FD3(747)
EQUIVALENCE(FD1(4),US(1,1)),(FD2(4),UA(1,1)),(FD3(4),UT(1,1))

(Subroutines you wish to call)

RETURN
END
```

(For SUM) - SUBROUTINE OPTION (MH,NIWDS,NRCDZ,ISAV,NECS,IGRD,JJ,II,JDATE)

```
DIMENSION US(45,56),UA(45,56),UT(45,56)
DIMENSION MH(NIWDS),ISAV(NIWDS,NECS)
COMMON/BLOCKA/FD1(2523),FD2(2523),FD3(2523)
EQUIVALENCE (FD1(4),US(1,1)),(FD2(4),UA(1,1)),(FD3(4),UT(1,1))

(Subroutines you wish to call)

RETURN
END
```

Subroutine SMTH (MH,NIWDS,NRCDZ,JJ,II,US,FD1,JDENT,IGRD,UA,FD2,IHR,JDATE,
MTH,ISAV,NECS)

This routine smooths field with identifier JDENT using MTH-point smoother where MTH=5, 9 or 25. All basic fields (fields from grid point tape) will automatically be smoothed and saved for interpolation by the program without having to call this routine in OPTION, if the smoothed identifier and processing indicator are properly defined on card type 4. When smoothing a computed field, you may smooth from OPTION or call this routine directly in your own subroutine. JDENT is dimensioned 3. JDENT(1) is set to first word ID of field to be smoothed, example (Z231022). JDENT(2) is set to second word ID, example (Z00001018) which indicates TJ field with TAU 24, JDENT(3) is set equal to JDATE. MTH is the smoother, Example 9. The smoothed field, in this case, would be stored with the MOS identiferZ231022,Z00201018. JDENT(1), JDENT(2) and MTH must be specified. All other arguments should just be passed from OPTION.

Subroutine RPECS(JDENT,JDATE,MH,NIWDS,FD1,NRCDZ,ISAV,NECS,JJ,ICND)

This routine searches memory for the packed field with identifier JDENT(1),JDENT(2), and date of JDATE. The search begins with the JJ field (JJ can be set to 1 if you wish to check through all the fields) and when located the field is unpacked and stored (predictor identifier, date (year, month, day, hour), and grid point values) in array FD1. ICND is determined by the RPECS and signifies the following:

(0) Field not saved
(1) Basic field saved for computation only.
(2) Computed field saved for interpolation. It may be used in computing additional computed fields.
(3) Basic field saved for interpolation. It may be used in computing new fields.
(4) This field not to be used for computation, station values set to 9999.
(5) Computed field saved for computation only.
(6) These values are read from cards and are valid at specified stations. The values will be saved on output tape as is. An example of a field that could have ICND=6 is station elevation.

If calling RPECS from one of your own subroutines as part of OPTION, you must define JDENT(1), JDENT(2) and JJ. ICND should be checked upon returning from the routine. Subroutine RCHECK may be used. The values of JDATE,MH,NIWDS,NRCDZ,ISAV and NECS are in the arguments of subroutine OPTION and should be carried through the arguments in your subroutine to RPECS without change.

EXAMPLE DERIVED PREDICTOR SUBROUTINE:

The following subroutine computes the wind speed at any level from the wind components and saves that field smoothed and/or unsmoothed:

```
      SUBROUTINE SPDBL(ID1,ID2,IHR,MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,JJ,II,
     1                 FD1,US,FD2,UA,FD3,UT,M0,M5,M9,M25)
C        SUBROUTINE COMPUTES WINDSPEED FROM COMPONENTS AND SAVES
C        RESULTING FIELD UNSMOOTHED AND/OR SMOOTHED.
C        FD1, FD2, AND FD3 ARE EQUIVALENCED TO US, UA, AND UT IN THE
C        CALLING SUBROUTINE OPTION IF EQUIVALENCE(FD1(1),US(1,1)),..,
C                 ID1- 1ST WORD ID OF PREDICTOR IF IF IN CALLING THIS
C                      ROUTINE ID1 IS SET TO 225062HEX, THE WIND SPEED
C                      WILL BE COMPUTED AT THE 225 LEVEL(BOUNDARY LEVEL)
C                      AND WRITTEN TO TAPE WITH 1ST WORD ID OF 22506/HEX.
C                 ID2- 2ND WORD ID OF PREDICTOR SHIFTED RIGHT 3 HEX
C                      PLACES IF FOR A 5 POINT SMOOTHED BOUNDARY LAYER
C                      WIND FROM THE LFM MODEL, ID2 IS SET TO 0102HEX.
C                 IHR- TAU
C                  M0- GREATER THAN 0 IF FIELD IS TO BE SAVED UNSMOOTHED.
C                  M5- GREATER THAN 0 IF FIELD  TO BE SAVED 5 PT SMOOTHED.
C                  M9- GREATER THAN 0 IF FIELD  TO BE SAVED 9 PT SMOOTHED.
C                 M25- GREATER THAN 0 IF FIELD TO BE SAVED 25 PT SMOOTHED.
      DIMENSION FD1(NRCDZ),US(JJ,II),FD2(NRCDZ),UA(JJ,II),FD3(NRCDZ),
     1          UT(JJ,II)
      DIMENSION IDENT(3)
      DIMENSION MH(NIWDS),ISAV(NIWDS,NECS)
      DATA IWND/7070/
C        LEVEL TAKEN FROM ID1 AND EAST-WEST WIND ATTACHED
      IDENT(1)=LOR(SHFTL(SHFTR(ID1,12),12),IWND)
C        COMPUTE 2ND WORD ID WITHOUT SMOOTHER INDICATORS
      IDENT(2)=SHFTR(SHFTL(ID2,24),12)+IHR
C        SAVE SECOND WORD ID IE MODEL NUMBER
      IDD=IDENT(2)
      IDENT(3)=JDATE
C        CHECK FOR SMOOTHER INDICATOR IN ID2
      MTH=SHFTR(SHFTL(ID2,20),28)
      N=1
C        READ EAST-WEST COMPONENT INTO FD1
      CALL RPECS(IDENT,JDATE,MH,NIWDS,FD1,NRCDZ,ISAV,NECS, N,ICND)
C        CHECK TO SEE THAT FIELD WAS FOUND
      CALL RCHECK(IDENT,JDATE,ICND)
C        SET TO FOR NORTH-SOUTH COMPONENT IE 071
      IDENT(1)=IDENT(1)+1
      N=1
C        READ NORTH SOUTH COMPONENT INTO FD2
      CALL RPECS(IDENT,JDATE,MH,NIWDS,FD2,NRCDZ,ISAV,NECS, N,ICND)
C        CHECK TO SEE THAT FIELD WAS FOUND
      CALL RCHECK(IDENT,JDATE,ICND)
C        COMPUTE SPEED FIELD AND STORE IN FD3
      DO 111 M=4,NRCDZ
      SQRU=FD1(M)*FD1(M)
      SQRV=FD2(M)*FD2(M)
      SSQRUV=SQRU+SQRV
      FD3(M)=SQRT(SSQRUV)
  111 CONTINUE
C        SET TO FOR SPEED FIELD
      IDENT(1)=ID1
      N=1
C        CHECK IF FIELD IS TO BE SAVED UNSMOOTHED
      IF(M0.EQ.0.AND.MTH.NE.6)GO TO 120
C        IF SO, WRITE FD3 TO TAPE
      CALL WECS(IDENT,JDATE,MH,NIWDS,FD3,NRCDZ,ISAV,NECS, N,ICND)
C        CHECK WRITE CONDITION CODE
      CALL WCHECK(IDENT,JDATE,ICND)
C        CHECK TO SEE IF SPEED FIELD IS TO BE SAVED 5 PT SMOOTHED.
  120 IF(MTH.NE.1.AND.M5.EQ.0)GO TO 130
      CALL SMTH5(JJ,II,UT,UA)
      IDENT(2)=LOR(IDD,SHFTL(1,20))
      ASSIGN 130 TO NUMBER
      GO TO 160
C        CHECK TO SEE IF SPEED FIELD IS TO BE SAVED 9 PT SMOOTHED.
  130 IF(MTH.NE.2.AND.M9.EQ.0)GO TO 140
      CALL SMTH9(JJ,II,UT,UA)
      IDENT(2)=LOR(IDD,SHFTL(2,20))
      ASSIGN 140 TO NUMBER
      GO TO 160
C        CHECK TO SEE IF SPEED FIELD IS TO BE SAVED 25 PT SMOOTHED.
  140 IF(MTH.NE.3.AND.M25.EQ.0)GO TO 170
      CALL SMTH25(JJ,II,UT,UA)
      IDENT(2)=LOR(IDD,SHFTL(3,20))
      ASSIGN 170 TO NUMBER
C        SMOOTHED FIELD WRITTEN AND CHECKED
  160 CALL WECS(IDENT,JDATE,MH,NIWDS,FD2,NRCDZ,ISAV,NECS,N,ICND)
      CALL WCHECK(IDENT,JDATE,ICND)
C        CONTINUE CHECKING DIFFERENT WAYS FIELD IS TO BE SAVED
      GO TO NUMBER,(130,140,170)
  170 RETURN
      END
```

JJ,II - Grid dimensions carried thru OPTION, i.e., for TJ JJ=13, II=17

ID1, ID2, IHR and any additional arguments not shown above must be supplied by the programmer. The name of the routine, XXXX, is also supplied by the programmer. The arguments MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,JJ,II,FD1,US,FD2, UA,FD3, and UT should be passed, without change, from subroutine OPTION.

In order for the same subroutine used in the development to be used in the operational program, the programmer must insure that, no matter the number of different fields that he computes in a single subroutine, he makes provisions, through the arguments, that any one of the fields, unsmoothed, can be computed seperately. The easiest way to do this is to add arguments to the general form shown above which indicate which of the different fields are to be computed. The routine that follows gives an example of this. To the extent practicable, each derived predictor should be computed in a seperate subroutine.

MOS OP NO. 19

MOSCST

RETRIEVES MOS FORECAST DATA

Frederick Marshall
November 30, 1974

PURPOSE: This subroutine is designed to retrieve MOS forecasts and/or station constants from the MOSMAT permanent files.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
DIMENSION FLD(350)
.
.
CALL MOSCST(NCODE,NPROJ,INIT,FLD,NST,NCAT,NDATE,NERR
.
```

NCODE - MOS code number of data type requested. Codes $\leq$ 16 indicate station constant data; codes $\geq$ 100 indicate various forecast fields. (See Attachment 1.)

NPROJ - Projection of forecast field wanted in hours. Contains dummy parameter if constant data is requested.

INIT Initial time, or cycle, in GMT (00 or 12).

FLD( ) - Array to contain data that is returned. Data for any station occupies same position in all returned arrays. All values returned are REAL*4 except MOS Codes 1 (station WBAN numbers) and 2 (block/station numbers) which are INTEGER*4.

NST - Number of values returned in FLD( ).

NCAT - A number which distinguishes among probabilities and/or categorical forecasts. If a single equation produces a categorical forecast or single probability, NCAT=1. If a set of 5 equations produces 5 probabilities, NCAT= 1,2,3,4, and 5. If a categorical forecast is also added to these 5 probabilities it will have NCAT=6. Contains dummy parameter if constant data are requested.

MOS OP NO. 20

SPDBL

COMPUTES WIND SPEED

Frank T. Globokar
December 31, 1974

PURPOSE: Computes wind speed from wind components and saves resulting field unsmoothed and/or smoothed. This routine was written to be used with the OPTION subroutine of any of the M200 interpolation programs.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
.
SUBROUTINE SPDBL(ID1,ID2,IHR,MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,JJ,
     II,FD1,US,FD2,UA,FD3,UT,M0,M5,M9,M25)
```

ID1 - First 6 hex characters of wind speed field identification (see "Format of MOS Interpolated Tapes") written to tape. The level at which the wind speed is to be computed is taken from ID1. For example, an ID1 of 205062 hex would result in the 205 level (or 850 mb) wind speed being computed.

ID2 - Last 4 hex characters of wind speed field identification (see "Format of MOS Interpolated Tapes"). For example, a 5-pt. smoothed LFM field would have an ID2 of 0102 hex.

IHR - Forecast projection tau.

M0 - if greater than 0, field is saved unsmoothed.

M5 - if greater than 0, field is saved 5-pt. smoothed.

M9 - if greater than 0, field is saved 9-pt. smoothed.

M25 - if greater than 0, field is saved 25-pt. smoothed.

MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,JJ,II,FD1,US,FD2,UA,FD3,UT - These arguments should be carried from OPTION to SPDBL without change.

EXAMPLE:

FOR LFM -
```
SUBROUTINE OPTION (MH,NIWDS,NRCDZ,ISAV,NECS,IGRD,JJ,II,JDATE)
DIMENSION US(24,31),UA(24,31),UT(24,31)
DIMENSION MH(NIWDS),ISAV(NIWDS,NECS)
```

DIVG

COMPUTES WIND DIVERGENCE

Frank T. Globokar
December 31, 1974

PURPOSE: Computes wind divergence from wind components and saves resulting field unsmoothed and/or smoothed. This routine was written to be used with the OPTION subroutine of any of the M200 interpolation programs.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
SUBROUTINE DIVG(ID1,ID2,IHR,MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,
   JJ,II,FD1,US,FD2,UA,FD3,UT,MO,M5,M9,M25)
```

ID1 - First 6 hex characters of divergence identification (see "Format of MOS Interpolated Tapes") as it is to be written to tape. The level at which the divergence is to be computed is taken from ID1. For example, an ID1 of 205112 hex would result in the 205 level (or 850 mb) divergence being computed.

ID2 - Last 4 characters of divergence identification (see "Format of MOS Interpolated Tapes"). For example, a 5-pt. smoothed PE field would have an ID2 of 0100 hex.

IHR - Forecast projection tau.

MO - If greater than 0, field is saved unsmoothed.

M5 - If greater than 0, field is saved 5-pt. smoothed.

M9 - If greater than 0, field is saved 9-pt. smoothed.

M25 - If greater than 0, field is saved 25-pt. smoothed.

MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,JJ,II,FD1,US,FD2,UA,FD3,UT - These arguments should be carried from OPTION to DIVG without change.

MOS OP NO. 22

VORT

COMPUTES VORTICITY

Frank T. Globokar
December 31, 1974

PURPOSE: Computes vorticity from forecast wind components or from geostrophic winds and saves resulting field unsmoothed and/or smoothed. This routine was written to be used with the OPTION subroutine of any of the M200 interpolation programs.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
SUBROUTINE VORT(ID1,ID2,IHR,MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,JJ,II,FD1,
    US,FD2,UA,FD3,UT,IVORT,M0,M5,M9,M25)
```

ID1 - First 6 hex characters of vorticity identification (see "Format of MOS Interpolated Tapes") as it is to be written to tape. The level at which the wind speed is to be computed is taken from ID1. For example, an ID1 of 205111 hex would result in the 205 level (or 850 mb) vorticity being computed.

ID2 - Last 4 hex characters of vorticity identification (see "Format of MOS Interpolated Tapes"). For example, a 5-pt. smoothed LFM field would have an ID2 of 0102 hex.

IHR - Forecast projection tau.

IVORT - Set equal to 0 if vorticity is to be computed from forecast winds. Set equal to 1 if vorticity is to be computed from geostrophic winds.

M0 - If greater than 0, field is saved unsmoothed.

M5 - If greater than 0, field is saved 5 pt. smoothed.

M9 - If greater than 0, field is saved 9 pt. smoothed.

M25 - If greater than 0, field is saved 25 pt. smoothed.

MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,JJ,II,FD1,US,FD2,UA,FD3,UT - These arguments should be carried from OPTION to SPDBL without change.

MOS OP NO. 23

DIFF

COMPUTES DIFFERENCE

Frank T. Globokar
December 31, 1974

PURPOSE: Computes difference between two fields and saves resulting field unsmoothed and/or smoothed. This routine was written to be used with the OPTION subroutine of any of the M200 interpolation programs.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
SUBROUTINE DIFF(J1,JHR,K1,KHR,KA1,KA5,KA9,KA25,JDATE,MH,
  NIWDS,NRCDZ,ISAV,NECS,US,UA,UT,JJ,II,FD1,FD2,FD3,IGRD)
```

J1 - 1st word ID (see "Format of MOS Interpolated Tapes") of minuend field.

JHR - Tau of minuend field.

K1 - 1st word ID of subtrahend field.

KHR - Tau of subtrahend field.

KA1 - 1st word ID of unsmoothed remainder field. If KA1 is set to 0, the unsmoothed remainder field will not be saved.

KA5 - 1st word ID of 5-pt. smoothed remainder field. If KA5 is set to 0, the 5-pt. smoothed remainder field will not be saved.

KA9 - 1st word ID of 9-pt. smoothed remainder field. If KA9 is set to 0, the 9-pt. smoothed remainder field will not be saved.

KA25 - 1st word ID of 25-pt. smoothed remainder field. If KA25 is set to 0, the 25-pt. smoothed remainder field will not be saved.

JDATE,MH,NIWDS,NRCDZ,ISAV,NECS,US,UA,UT,JJ,II,FD1,FD2,FD3,IGRD - These arguments should be carried from OPTION to DIFF without change.

FORIER

COMPUTES SIN DOY, COS DOY, SIN 2*DOY, OR COS 2*DOY

Frank T. Globokar
December 31, 1974

PURPOSE: Computes Sin $2\pi i/365$, Cos $2\pi i/365$, Sin $4\pi i/365$, and Cos $4\pi i/365$, where i = the day of the year (1=Jan. 1, etc.). These 4 variables are called SIN DOY, COS DOY, SIN 2*DOY, AND COS 2*DOY, respectively. This routine was written to be used with the OPTION subroutine of any of the M200 interpolation programs.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
SUBROUTINE FORIER(ID1,ID2,IHR,MH,NIWDS,FD1,NRCDZ,ISAV,NECS,JDATE)
```

ID1 - First 6 hex characters (see "Format of MOS Interpolated Tapes") of field to be computed. Set ID1 to 000400 hex for SIN DOY, 000401 for COS DOY, 000402 for SIN 2*DOY, or 000403 for COS 2*DOY.

ID2 - Last 4 characters (see "Format of MOS Interpolated Tapes") which, in this case, would be the model number.

IHR - Forecast projection tau.

MH,NIWDS,NRCDZ,ISAV,NECS,FD1,JDATE, - These arguments should be carried from OPTION to FORIER without change.

EXAMPLE:

(For PE) -

```
SUBROUTINE OPTION (MH,NIWDS,NRCDZ,ISAV,NECS,IGRD,JJ,II,JDATE)
DIMENSION US(15,17),AK(8,7),WH(4,3),PI(2,2)
DIMENSION UA(15,17),AS(8,7),WI(4,3),PR(2,2)
DIMENSION UT(15,17),AL(8,7),WA(4,3),PT(2,2)
DIMENSION MH(NIWDS),ISAV(NIWDS,NECS)
COMMON/BLOCKA/FD1(330),FD2(330),FD3(330)
EQUIVALENCE (FD1(4),US(1,1)),(FD1(259),AK(1,1)),(FD1(315),WH(1,1)),
    (FD1(327),PI(1,1))
EQUIVALENCE (FD2(4),UA(1,1)),(FD2(259),AS(1,1)),(FD2(315),WI(1,1)),
    (FD2(327),PR(1,1))
EQUIVALENCE (FD3(4),UT(1,1)),(FD3(259),AL(1,1)),(FD3(315),WA(1,1)),
    (FD3(327),PT(1,1))
.
.
CALL FORIER (IHX(000402),IHX(0000),00,MH,NIWDS,FD1,NRCDZ,ISAV,NECS,
      JDATE)
.
.
RETURN
END
```

GINDEX

COMPUTES G-INDEX

Frank T. Globokar
December 31, 1974

PURPOSE: Computes G-INDEX (defined by H. Glahn G=2*Z850-Z1000-Z500) and saves resulting field unsmoothed and/or smoothed. This routine was written to be used with the OPTION subroutine of any of the M200 interpolation programs.

CALL AND EXPLANATION OF FORMAL PARAMETERS:

```
.
.
SUBROUTINE GINDEX (ID1,ID2,IHR,MH,NIWDS,NRCDZ,ISAV,NECS,
   JDATE,JJ,II,FD1,US,FD2,UA,M0,M5,M9,M25)
```

ID1 - First 6 hex characters of G-INDEX identification (see "Format of MOS Interpolated Tapes") as it is to be written to tape.

ID2 - Last 4 hex characters of G-INDEX identification (see "Format of MOS Interpolated Tapes"). For example, a 5-pt. smoothed LFM field would have an ID2 of 0102 hex.

IHR - Forecast projection tau.

M0 - If greater than 0, field is saved unsmoothed.

M5 - If greater than 0, field is saved 5-pt. smoothed.

M9 - If greater than 0, field is saved 9-pt. smoothed.

M25 - If greater than 0, field is saved 25-pt. smoothed.

MH,NIWDS,NRCDZ,ISAV,NECS,JDATE,JJ,II,FD1,US,FD2,UA,FD3,UT - These arguments should be carried from OPTION to GINDEX without change.

EXAMPLE:

For LFM -

```
SUBROUTINE OPTION (MH,NIWDS,NRCDZ,ISAV,NECS,IGRD,JJ,II,JDATE)
DIMENSION US(24,31),UA(24,31),UT(24,31)
DIMENSION MH(NIWDS),ISAV(NIWDS,NECS)
COMMON/BLOCKA/FD1(747),FD2(747),FD3(747)
EQUIVALENCE(FD1(4),US(1,1)),(FD2(4),UA(1,1)),(FD3(4),UT(1,1))
```